

# INFORME DE LABORATORIO

**Autores:** *Angee Lorena Ocampo Ramírez, Oscar Andrés Gutiérrez Rivadeneira*

*Laboratorio de Electrónica Digital 2  
Departamento de Ingeniería Electrónica y de Telecomunicaciones  
Universidad de Antioquia*

## Resumen

Durante la presente práctica se desarrolló una máquina de estados que debía seguir una secuencia dada mediante el lenguaje de descripción de hardware, SystemVerilog, siendo puesta a prueba mediante la herramienta de simulación ModelSim, e implementada en la FPGA DE10-lite; para lograr este objetivo, se hizo un análisis previo que ayudara a segmentar el problema y determinar los módulos a implementar. Se identificó que la manera más sencilla de elaborar esta práctica era mediante un contador que posibilitaba el cambio de estado de manera síncrona, como se verá en el desarrollo del presente informe.

**Palabras clave:** Contador, FPGA, Máquina de estados, SystemVerilog

## Procedimiento experimental y resultados

A continuación se procede a describir el análisis realizado para el desarrollo de la presente práctica, también, se mostrará el diseño desarrollado y la simulación del mismo.

### Análisis:

Inicialmente, se identificó que eran necesarios cuatro módulos aparte de la entidad principal para la ejecución del diseño requerido. La conexión de los mismos es la vista en la figura 0-1.

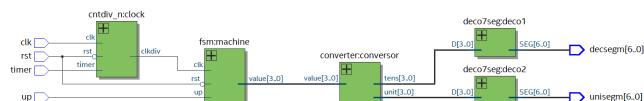


Figura 0-1: Diagrama de bloques

El primero de estos es el módulo que genera los relojes, este recibe como entrada el timer, que es la bandera que indica cuál de los dos debe ser generado con base en el reloj de la tarjeta, la salida de esta entidad será el reloj requerido.

El segundo módulo es la máquina de estados (ver imagen 0-2), que en este caso fue empleada mediante un arreglo que contenía la secuencia de datos, al cual se accede mediante la posición; el acceso se efectuara de manera ascendente si la bandera up, que es una entrada, está en uno, o descendente si esta es cero; es importante resaltar, que la FMS cambia su estado de acuerdo al reloj generado en el primer módulo, es decir, que en cada flanco de subida, producido cada 0.5s o 1s de acuerdo al timer, que es otra entrada, este actualiza su estado; asimismo, esta entidad contiene un reset que reinicia el recorrido sobre el arreglo.

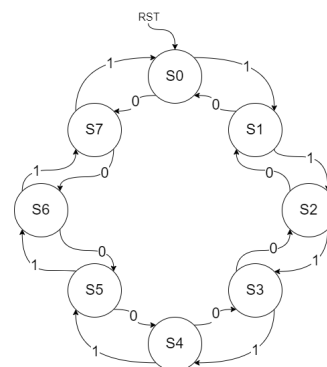


Figura 0-2: Máquina de estados

El tercer módulo elaborado tiene como entrada el valor de la secuencia que es dado por el módulo anterior, en esta entidad el valor será separado entre sus decenas y unidades con el propósito de mostrarlos en el display de siete segmentos de la tarjeta; por lo anterior, durante el proceso combinacional es utilizada la operación módulo para extraer las unidades y la división para obtener las decenas. La cuarta entidad elaborada es el decodificador de siete segmentos, que recibe el número en su codificación binaria, para posteriormente hacer su respectiva conversión con el fin de que el número ingresado sea mostrado en el display.

Por último, se tiene la entidad principal, en esta se realizan todas las conexiones entre los módulos, y aquí se definen los parámetros, además de las respectivas entradas y salidas requeridas.

### Simulación

Junto con la entidad principal, también fue desarrollado el test bench con el que se ejecutó la simulación a través de la herramienta ModelSim, como se puede observar en las figuras 0-3 y 0-4.

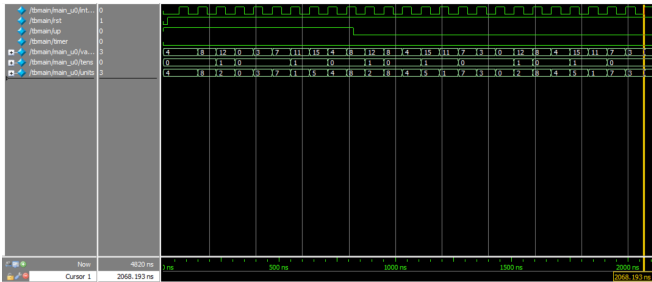


Figura 0-3: Caption

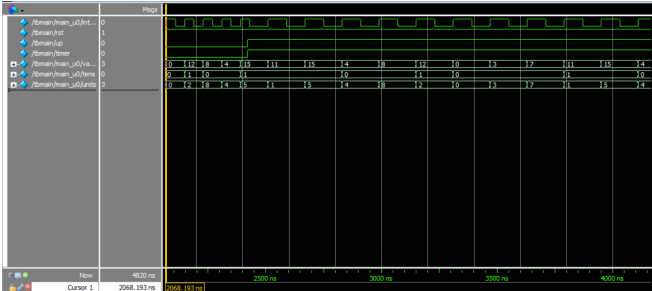


Figura 0-4: Caption

En la simulación mostrada se observa todos los relojes

del diseño, que son el reloj de la tarjeta y los dos generados, las banderas up y timer, que determinan el orden de la secuencia y el tiempo con el cual se hará el cambio de estado; de igual modo, se muestra el valor de la secuencia y su decodificación para el display de siete segmentos. En el test bench, se redujeron los contadores de los relojes con la intención de reducir el tiempo de simulación, de forma que con pocos ciclos se pueda apreciar el comportamiento del hardware diseñado.

### Conclusiones

Si bien el objetivo de la práctica era el de realizar una máquina de estados, el uso de un contador simplifica el código, reduciendo el tamaño y facilitando su comprensión.

Por otra parte, el orden en que se afronte el problema permite facilitar ciertos procesos, esto debido a que el análisis previo, juntos con los diagramas realizados, permite tener una ruta de trabajo clara, permitiendo que las horas de trabajo sean efectivas. Cabe resaltar que el tiempo invertido en esta actividad fue de aproximadamente 9 horas.